Package 'BayesSPsurv'

September 13, 2021

Type Package

Title Bayesian Spatial Split Population Survival Model

Version 0.1.4

Description Parametric spatial split-population (SP) survival models for clustered
event processes. The models account for structural and spatial heterogeneity among
"at risk" and "immune" populations, and incorporate time-varying covariates.
This package currently implements Weibull, Exponential and Log-logistic forms for
the duration component. It also includes functions for a series of diagnostic
tests and plots to easily visualize spatial autocorrelation, convergence, and spatial ef-
fects. Users can
create their own spatial weights matrix based on their units and adjacencies of
interest, making the use of these models flexible and broadly applicable to a
variety of research areas. Joo et al. (2020) <`https://github.com/Nicolas-Schmidt/`
`BayesSPsurv/blob/master/man/figures/SPcure.pdf`> describe
the estimators included in this package.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 3.6.0)

BugReports `https://github.com/Nicolas-Schmidt/BayesSPsurv/issues`

URL `https://nicolas-schmidt.github.io/BayesSPsurv/`

RoxygenNote 7.1.1

LinkingTo Rcpp, RcppArmadillo

Imports MCMCpack, FastGP, stats, Rcpp (>= 1.0.3), coda, dplyr,
reshape2, ggplot2, ape, progress, rworldmap, countrycode

Suggests spduration

NeedsCompilation yes

Author Brandon L. Bolte [aut],
Nicolas Schmidt [aut, cre],
Sergio Bejar [aut],
Minnie M. Joo [aut],

Nguyen K. Huynh [aut],
Bumba Mukherjee [aut]

R topics documented:

---

bsps_map	*bsps_map*

Description

Implement the `mapCountryData` function in a short and fast version.

Usage

```
bsps_map(data, mapTitle = "columnName")
```

Arguments

data	data.
mapTitle	title to add to the map, any string or 'columnName' to set it to the name of the data column.

Examples

```
library(BayesSPsurv)

walter <- spduration::add_duration(Walter_2015_JCR,"renewed_war",
                                   unitID = "id", tID = "year",
                                   freq = "year", ongoing = FALSE)
```

```
walter <- BayesSPsurv::spatial_SA(data = walter, var_ccode = "ccode", threshold = 800L)

set.seed(123456)

model <- spatialSPsurv(
    duration  = duration ~ victory + comprehensive + lgdpl + unpko,
    immune    = atrisk ~ lgdpl,
    Y0        = 't.0',
    LY        = 'lastyear',
    S         = 'sp_id' ,
    data      = walter[[1]],
    N         = 1500,
    burn      = 300,
    thin      = 15,
    w         = c(1,1,1),
    m         = 10,
    form      = "Weibull",
    prop.varV = 1e-05,
    prop.varW = 1e-03,
    A         = walter[[2]]
)

bsps_map(data = model$W)
```

---

capdist   *Gleditsch and Ward Distance data*

Description

Dyadic dataset extracted from Gleditsch and Ward (2001). The dataset contains information on the distace between capital cities among independent nation-states.

Usage

```
data(capdist)
```

Format

A data frame with 41006 rows and 6 variables

Details

numa COW code – country A.

ida Three letter ISO code – country A.

numb COW code – country B.

idb Three letter ISO code – country B.

kmdist Distance between capital cities in the kilometers.

midist Minimal distance between capital cities in the kilometers.

Source

Gleditsch, Kristian S., and Michael D. Ward. (2001). "Measuring Space: A Minimum-Distance Database and Applications to International Studies." Journal of Conflict Resolution 38(6): 739-758.

---

exchangeSPsurv   *exchangeSPsurv*

Description

Markov Chain Monte Carlo (MCMC) to run Bayesian split population survival model with exchangeable frailties.

Returns a summary of a exchangeSPsurv object via `summary.mcmc`.

Print method for a `exchangeSPsurv` x.

Returns a plot of a exchangeSPsurv object via `plot.mcmc`.

Usage

```
exchangeSPsurv(
  duration,
  immune,
  Y0,
  LY,
  S,
  data,
  N,
  burn,
  thin,
  w = c(1, 1, 1),
  m = 10,
  ini.beta = 0,
  ini.gamma = 0,
  ini.W = 0,
  ini.V = 0,
  form = c("Weibull", "exponential", "loglog"),
  prop.varV,
  prop.varW,
  id_WV = unique(data[, S])
)

## S3 method for class 'frailtySPsurv'
summary(object, parameter = character(), ...)

## S3 method for class 'frailtySPsurv'
print(x, ...)
```

```
## S3 method for class 'frailtySPsurv'
plot(x, parameter = character(), ...)
```

Arguments

duration	survival stage equation written in a formula of the form Y ~ X1 + X2 + ... where Y is duration until failure or censoring.
immune	split stage equation written in a formula of the form C ~ Z1 + Z2 + ... where C is a binary indicator of immunity.
Y0	the elapsed time since inception until the beginning of time period (t-1).
LY	last observation year (coded as 1; 0 otherwise) due to censoring or failure.
S	spatial information (e.g. district ID) for each observation that matches the spatial matrix row/column information.
data	data.frame.
N	number of MCMC iterations.
burn	burn-in to be discarded.
thin	thinning to prevent from autocorrelation.
w	size of the slice in the slice sampling for (betas, gammas, rho). Write it as a vector. E.g. c(1,1,1).
m	limit on steps in the slice sampling. A vector of values for beta, gamma, rho.
ini.beta	initial value for the parameter vector beta. By default is 0.
ini.gamma	initial value for the parameter vector gamma. By default is 0.
ini.W	initial value for the parameter vector W. By default is 0.
ini.V	initial value for the parameter vector V. By default is 0.
form	type of parametric model (Weibull, Exponential, or Log-Logistic).
prop.varV	proposed variance for V in Metropolis-Hastings.
prop.varW	proposed variance for W in Metropolis-Hastings.
id_WV	vector of type character that modifies the colnames of W and V in the model's result. By default is unique(data[,S]).
object	an object of class frailtySPsurv, the output of [exchangeSPsurv]().
parameter	one of five parameters of the [spatialSPsurv]() output. Indicate either "betas," "gammas," "rho", "lambda" or "delta".
...	additional parameter.
x	an object of class frailtySPsurv, the output of [exchangeSPsurv]().

Value

exchangeSPsurv returns an object of class "frailtySPsurv".

An "exchangeSPsurv" object has the following elements:

betas	matrix, numeric values of the posterior for each variable in the duration equation.

gammas	matrix, numeric values of the posterior for each variable in the immune equation.
rho	vector, numeric values of rho.
lambda	vector, numeric values of lambda.
delta	vector, numeric values of delta.
W	matrix, numeric values of the posterior for Ws.
V	matrix, numeric values of the posterior for Vs.
X	matrix of X's variables.
Z	matrix of Z's variables.
Y	vector of 'Y'.
Y0	vector of 'Y0'.
C	vector of 'C'.
S	vector of 'S'.
ini.beta	numeric initial values of beta.
ini.gamma	numeric initial values of gamma.
ini.W	numeric initial values of W.
ini.V	numeric initial values of V.
form	character, type of distribution.
call	description for the model to be estimated.

list. Empirical mean, standard deviation and quantiles for each variable.

list. Empirical mean, standard deviation and quantiles for each variable.

Examples

```
## 1

walter <- spduration::add_duration(Walter_2015_JCR,"renewed_war",
                                   unitID = "ccode", tID = "year",
                                   freq = "year", ongoing = FALSE)

# add S
walter <- spatial_SA(data = walter, var_ccode = "ccode", threshold = 800L)

set.seed(123456)

model <-
    exchangeSPsurv(
         duration  = duration ~ fhcompor1 + lgdpl + comprehensive + victory +
                     instabl + intensityln + ethfrac + unpko,
         immune    = cured ~ fhcompor1 + lgdpl + victory,
         Y0        = 't.0',
         LY        = 'lastyear',
```

```
            S         = 'sp_id' ,
            data      = walter[[1]],
            N         = 100,
            burn      = 10,
            thin      = 10,
            w         = c(1,1,1),
            m         = 10,
            form      = "Weibull",
            prop.varV = 1e-05,
            prop.varW = 1e-05,
)

print(model)

summary(model, parameter = "betas")

# plot(model)


## 2

walter   <- spduration::add_duration(Walter_2015_JCR,"renewed_war",
                                 unitID = "ccode", tID = "year",
                                 freq = "year", ongoing = FALSE)

walter$S <- rep(x = 1:length(unique(walter$ccode)), times = rle(walter$ccode)$lengths)
country  <- countrycode::countrycode(unique(walter$ccode),'gwn','iso3c')

set.seed(123456)

model <-
    exchangeSPsurv(
        duration  = duration ~ fhcompor1 + lgdpl + comprehensive + victory +
                     instabl + intensityln + ethfrac + unpko,
        immune    = cured ~ fhcompor1 + lgdpl + victory,
        Y0        = 't.0',
        LY        = 'lastyear',
        S         = 'S' ,
        data      = walter,
        N         = 100,
        burn      = 10,
        thin      = 10,
        w         = c(1,1,1),
        m         = 10,
        form      = "loglog",
        prop.varV = 1e-05,
        prop.varW = 1e-05,
        id_WV     = country
    )

print(model)
```

plot_JoinCount	*plot_JoinCount*

Description

Uses Join Count tests to assess spatial clustering or dispersion of categorical variables in the data. Negative values indicate positive spatial clustering.

Usage

```
plot_JoinCount(
  data,
  var_cured = character(),
  var_id = character(),
  var_time = character(),
  n = 1,
  t = 1.645,
  threshold = 800L
)
```

Arguments

data	data.
var_cured	binary indicator of immunity.
var_id	ID's unique identifier.
var_time	variable that measures time.
n	number of observation per id.
t	value of the confidence interval.
threshold	distance in kilometers. By default is 800.

Value

A ggplot object

Examples

```
library(BayesSPsurv)
dataw  <- spduration::add_duration(data = BayesSPsurv::Walter_2015_JCR,
                                   y = "renewed_war",
                                   unitID = "ccode",
                                   tID = "year",
                                   freq = "year",
                                   ongoing = FALSE)
```

```
plot_JoinCount(data = dataw,
               var_cured = "cured",
               var_id = "ccode",
               var_time = "year",
               n = 12)
```

```
plot_Moran.I                    plot_Moran.I
```

Description

Implements Global Moran I test to evaluate spatial autocorrelation in a units' risk propensity in the data. Positive values indicate spatial clustering of similar values.

Usage

```
plot_Moran.I(
  data,
  var_duration = character(),
  var_id = character(),
  var_time = character(),
  n = 1,
  t = 1.645,
  threshold = 800L
)
```

Arguments

data	data.
var_duration	variable that measures duration until censoring or failure.
var_id	ID's unique identifier.
var_time	variable that measures time.
n	number of observation per id.
t	value of the confidence interval.
threshold	distance in kilometers. By default is 800.

Value

A ggplot object

Examples

```
library(BayesSPsurv)
dataw <- spduration::add_duration(data = BayesSPsurv::Walter_2015_JCR,
                                  y = "renewed_war",
                                  unitID = "ccode",
                                  tID = "year",
                                  freq = "year",
                                  ongoing = FALSE)


plot_Moran.I(data = dataw ,
             var_duration = "duration",
             var_id = "ccode",
             var_time = "year",
             n = 12)
```

---

pooledSPsurv   *pooledSPsurv*

Description

Markov Chain Monte Carlo (MCMC) to run Bayesian split population survival model with no frailties.

Returns a summary of a SPsurv object via `summary.mcmc`.

Print method for a `pooledSPsurv` x.

Returns a plot of a pooledSPsurv object via `plot.mcmc`.

Usage

```
pooledSPsurv(
  duration,
  immune,
  Y0,
  LY,
  data,
  N,
  burn,
  thin,
  w = c(1, 1, 1),
  m = 10,
  ini.beta = 0,
  ini.gamma = 0,
  form = c("Weibull", "exponential", "loglog")
)
```

```
## S3 method for class 'SPsurv'
summary(object, parameter = character(), ...)

## S3 method for class 'SPsurv'
print(x, ...)

## S3 method for class 'SPsurv'
plot(x, parameter = character(), ...)
```

Arguments

duration	survival stage equation written in a formula of the form Y ~ X1 + X2 + ... where Y is duration until failure or censoring.
immune	split stage equation written in a formula of the form C ~ Z1 + Z2 + ... where C is a binary indicator of immunity.
Y0	the elapsed time since inception until the beginning of time period (t-1).
LY	last observation year (coded as 1; 0 otherwise) due to censoring or failure.
data	data.frame.
N	number of MCMC iterations.
burn	burn-in to be discarded.
thin	thinning to prevent from autocorrelation.
w	size of the slice in the slice sampling for (betas, gammas, rho). Write it as a vector. E.g. c(1,1,1).
m	limit on steps in the slice sampling. A vector of values for beta, gamma, rho.
ini.beta	initial value for the parameter vector beta. By default is 0.
ini.gamma	initial value for the parameter vector gamma. By default is 0.
form	type of parametric model (Weibull, Exponential, or Log-Logistic).
object	an object of class SPsurv, the output of [pooledSPsurv](#).
parameter	one of Four parameters of the [pooledSPsurv](#) output. Indicate either "betas," "gammas", "rho" or "delta".
...	additional parameter.
x	an object of class SPsurv, the output of [pooledSPsurv](#).

Value

pooledSPsurv returns an object of class "SPsurv".

A "pooledSPsurv" object has the following elements:

betas	matrix, numeric values of the posterior for each variable in the duration equation .
gammas	matrix, numeric values of the posterior for each variable in the immune equation.
rho	vector, numeric values of rho.

delta	vector, numeric values of delta.
X	matrix of X's variables.
Z	matrix of Z's variables.
Y	vector of 'Y'.
Y0	vector of 'Y0'.
C	vector of 'C'.
ini.beta	numeric initial value of beta.
ini.gamma	numeric initial value of gamma.
form	character, type of distribution.
call	description for the model to be estimated.

list. Empirical mean, standard deviation and quantiles for each variable.

list. Empirical mean, standard deviation and quantiles for each variable.

Examples

```
walter <- spduration::add_duration(Walter_2015_JCR,"renewed_war",
                                   unitID = "ccode", tID = "year",
                                   freq = "year", ongoing = FALSE)

set.seed(123456)

model <-
    pooledSPsurv(
        duration = duration ~ fhcompor1 + lgdpl + comprehensive + victory +
                   instabl + intensityln + ethfrac + unpko,
        immune   = cured ~ fhcompor1 + lgdpl + victory,
        Y0       = 't.0',
        LY       = 'lastyear',
        data     = walter,
        N        = 100,
        burn     = 10,
        thin     = 10,
        w        = c(1,1,1),
        m        = 10,
        form     = "Weibull"
    )


print(model)

summary(model, parameter = "betas")

# plot(model)
```

---

spatialSPsurv   *spatialSPsurv*

Description

Markov Chain Monte Carlo (MCMC) to run time-varying Bayesian split population survival model with spatial frailties.

Returns a summary of a exchangeSPsurv object via `summary.mcmc`.

Print method for a `spatialSPsurv` x.

Returns a plot of a spatialSPsurv object via `plot.mcmc`.

Usage

```
spatialSPsurv(
  duration,
  immune,
  Y0,
  LY,
  S,
  A,
  data,
  N,
  burn,
  thin,
  w = c(1, 1, 1),
  m = 10,
  ini.beta = 0,
  ini.gamma = 0,
  ini.W = 0,
  ini.V = 0,
  form = c("Weibull", "exponential", "loglog"),
  prop.varV,
  prop.varW,
  id_WV = colnames(A)
)

## S3 method for class 'spatialSPsurv'
summary(object, parameter = character(), ...)

## S3 method for class 'spatialSPsurv'
print(x, ...)

## S3 method for class 'spatialSPsurv'
plot(x, parameter = character(), ...)
```

Arguments

duration	survival stage equation written in a formula of the form Y ~ X1 + X2 + ... where Y is duration until failure or censoring.
immune	split stage equation written in a formula of the form C ~ Z1 + Z2 + ... where C is a binary indicator of immunity.
Y0	the elapsed time since inception until the beginning of time period (t-1).
LY	last observation year (coded as 1; 0 otherwise) due to censoring or failure.
S	spatial information (e.g. district ID) for each observation that matches the spatial matrix row/column information.
A	an a times a spatial weights matrix where a is the number of unique spatial units (S) load as a separate file.
data	data.frame.
N	number of MCMC iterations.
burn	burn-in to be discarded.
thin	thinning to prevent from autocorrelation.
w	size of the slice in the slice sampling for (betas, gammas, rho). Write it as a vector. E.g. c(1,1,1).
m	limit on steps in the slice sampling. A vector of values for beta, gamma, rho.
ini.beta	initial value for the parameter vector beta. By default is 0.
ini.gamma	initial value for the parameter vector gamma. By default is 0.
ini.W	initial value for the parameter vector W. By default is 0.
ini.V	initial value for the parameter vector V. By default is 0.
form	type of parametric model (Weibull, Exponential, or Log-Logistic).
prop.varV	proposal for variance of V in Metropolis-Hastings.
prop.varW	proposal for variance of W in Metropolis-Hastings.
id_WV	vector of type character that modifies the colnames of W and V in the model's result. By default is colnames(A).
object	an object of class spatialSPsurv, the output of [spatialSPsurv]().
parameter	one of five parameters of the [spatialSPsurv]() output. Indicate either "betas," "gammas," "rho", "lambda" or "delta".
...	additional parameter.
x	an object of class spatialSPsurv, the output of [spatialSPsurv]().

Value

spatialSPsurv returns an object of class "spatialSPsurv".

A "spatialSPsurv" object has the following elements:

betas	matrix, numeric values of the posterior for each variable in the duration equation.
gammas	matrix, numeric values of the posterior for each variable in the immune equation.

rho	vector, numeric values of rho.
lambda	vector, numeric values of lambda.
delta	vector, numeric values of delta.
W	matrix, numeric values of the posterior for Ws.
V	matrix, numeric values of the posterior for Vs.
X	matrix of X's variables.
Z	matrix of Z's variables.
Y	vector of 'Y'.
Y0	vector of 'Y0'.
C	vector of 'C'.
S	vector of 'S'.
ini.beta	numeric initial values of beta.
ini.gamma	numeric initial values of gamma.
ini.W	numeric initial values of W.
ini.V	numeric initial values of V.
form	character, type of distribution.
call	description for the model to be estimated.

list. Empirical mean, standard deviation and quantiles for each variable.

list. Empirical mean, standard deviation and quantiles for each variable.

Examples

```
walter <- spduration::add_duration(Walter_2015_JCR,"renewed_war",
                                 unitID = "ccode", tID = "year",
                                 freq = "year", ongoing = FALSE)

walter <- spatial_SA(data = walter, var_ccode = "ccode", threshold = 800L)

set.seed(123456)

model <-
    spatialSPsurv(
        duration  = duration ~ fhcompor1 + lgdpl + comprehensive + victory +
                    instabl + intensityln + ethfrac + unpko,
        immune    = cured ~ fhcompor1 + lgdpl + victory,
        Y0        = 't.0',
        LY        = 'lastyear',
        S         = 'sp_id' ,
        data      = walter[[1]],
        N         = 100,
        burn      = 10,
        thin      = 10,
        w         = c(1,1,1),
        m         = 10,
```

```
        form     = "Weibull",
        prop.varV = 1e-05,
        prop.varW = 1e-05,
        A        = walter[[2]]
    )

print(model)

summary(model, parameter = "betas")

# plot(model)
```

---

spatial_SA   *spatial_SA*

Description

Generates a spatial weights matrix (A) and sp_id (S). User defines units and adjacencies.

Usage

```
spatial_SA(data, var_ccode, threshold = 800L)
```

Arguments

data	data.frame.
var_ccode	name of the variable that contains the country codes.
threshold	distance in kilometers. By default is 800.

Value

list. Contains database with variable sp_id (S) and matrix A.

Examples

```
walter <- spduration::add_duration(Walter_2015_JCR,"renewed_war",
                                   unitID = "ccode",
                                   tID = "year",
                                   freq = "year",
                                   ongoing = FALSE)

walter <- spatial_SA(data = walter,
                     var_ccode = "ccode",
                     threshold = 800L)
```

SPstats   *SPstats*

Description

Calculates the deviance information criterion (DIC) and Log-likelihood for fitted model outputs of pooled, exchangeable, and spatial Split Population survival models for which a log-likelihood can be obtained using the formula *DIC = -2 * (L - P)*, where *L* is the log likelihood of the data given the posterior means of the parameter and *P* is the estimate of the effective number of parameters in the model.

Usage

```
SPstats(object)
```

Arguments

object   An object of the output of pooled, exchangeable, or spatial Split Population survival model .

Value

List.

Walter_2015_JCR   *Walter_2015_JCR*

Description

Subsetted version of a time-series-cross-sectional (TSCS) dataset used in Walter (2015). It has data on duration of post-war peace as well as information on other relevant economic and political data.

Usage

```
data(Walter_2015_JCR)
```

Format

A data frame with 1237 rows and 13 variables

Details

year year.

lastyear last observation year.

renewed_war binary variable coded as 1 if the war was fought.

fhcompor1 Freedom House civil liberties index.

lgdpl log of per capita GDP in 2005 dollars.

comprehensive combatants signed comprehensive peace agreement.

victory end of previous war with outright victory.

instabl dummy that indicates whether there was a positive or negative change in the Polity 2 score in the previous country-year.

intensityln deaths per year – logged.

ethfrac index of ethnic fractionalization.

unpko number of UN peacekeepers on the ground.

ccode country code.

id unique conflict identifier.

Source

Walter, Barbara F. (2015), Why Bad Governance Leads to Repeat Civil War, Journal of Conflict Resolution 59(7), 1242 - 1272.

Index